Subject: Re: no backwards compatibility in IDL 5.6
Posted by JD Smith on Fri, 28 Feb 2003 17:52:48 GMT
View Forum Message <> Reply to Message

On Fri, 28 Feb 2003 10:47:52 -0700, David Fanning wrote:

> Pavel Romashkin (pavel_romashkin@hotmail.com) writes:
>
>> Why is EXECUTE used in this program? Why can't the value just be
>> returned from each CASE? Execute will slow it down and as far as I can
>> tell, does nothing special. There is no code that follows the CASE to
>> prevent you from returning at any point. Will it not compile in 5.4
>> with the extra keyword? I thought keyword mismatches are runtime
>> errors. Am I missing something?
>
> I don't know. I got so confused with the discussion yesterday I finally
> just said the hell with it and went back to bed. :-(
>
> Let's just say I had no idea so many people used the ATAN function.
>
> I'm totally confused about when things will compile and when they won't.
> The only thing I know for sure is they won't compile if they have to.
> For example, they would never compile if you were doing a demo in front
> of the new Vice President of the company.
>
> I think there must have been a change somewhere along the way (while we
> are on this subject). Because I didn't expect that file to compile in
> IDL 5.4, due to the REAL_PART function in the IDL 5.5 part of the CASE
> statement. When it did, that's when I realized I needed a nap.
>
>
I bet you didn't change your IDL PATH between running 5.5 and 5.4.
REAL_PART is in the !DIR/lib as a .pro file, and 5.4 can use it just as
well. Also, resolving a routine call into a compiled .pro file occurs at
run-time, so even if you called it REAL_PART_DOESNT_EXIST it would still
compile.

I.e.:

```
pro foo
  if 0 then MY_NON_EXISTENT_PROCEDURE,4
end
```

would compile and run perfectly fine.

IDL does check the number of arguments of *built-in* (i.e. not .pro)
system routines at compile time (this counts DLMs too). I think Pavel is
right that all keywords are checked at run-time.

JD